



Integer Programming for the Generalized (High) School Timetabling Problem

Kristiansen, Simon; Sørensen, Matias; Stidsen, Thomas Riis

Published in:

Proceedings of the 10th International Conference on the Practice and Theory of Automated Timetabling

Publication date:

2014

[Link back to DTU Orbit](#)

Citation (APA):

Kristiansen, S., Sørensen, M., & Stidsen, T. R. (2014). Integer Programming for the Generalized (High) School Timetabling Problem. In *Proceedings of the 10th International Conference on the Practice and Theory of Automated Timetabling*

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Integer Programming for the Generalized (High) School Timetabling Problem

Simon Kristiansen · Matias Sørensen · Thomas R. Stidsen

Received: date / Accepted: date

1 Introduction

The *High School Timetabling* (HSTT) is an important problem encountered at the high school in many countries. High School Timetabling is the problem of scheduling lectures to time slots and/or resources at high schools, and a large amount of different solution approaches have been proposed, see surveys Schaerf (1999); Kristiansen and Stidsen (2013)

It is well recognized that the specifications of the HST problem varies significantly depending on the country of which the problem originates, and that the problem in general is hard to solve.

Due to lack of exchangeable benchmark in a uniform format of the HSTT, Post et al (2012) introduced the XHSTT format, consisting of a large number of instances from various origins in standardized form. The format is based on the *Extensible Markup Language* (XML) standard, and all instances are available online (Post, 2013). One purpose of the format is to serve as a common test-bed for high school timetabling, in an attempt to promote research within this area.

In this paper, we will describe the first mixed *Mixed Integer linear Programming* (MIP) method capable of handling an arbitrary instance of the XHSTT format. The solution method is a two steps approach using a commercial general-purpose MIP solver. Computational results are performed for all the real-life instances currently available. By applying this MIP method, we are able to find previously unknown optimal solution, and prove optimality of already known solutions.

To the best of our knowledge, all previous methods applied for the XHSTT format have been heuristic in nature. Hence, no proof of optimality has been made for any instances, except for those instances where a solution with objective value 0 is known, as 0 is a trivial lower bound. Moreover, the quality of a given solution is rather hard to measure as only the trivial lower bound exist. An obvious advantage of *Integer Programming* (IP) is the capability to issue certificates of optimality. Therefore it is remarked that a big advance within general-purpose MIP solvers has happened in recent years, see e.g. Bixby (2012). Even though the created MIP is inevitable complex in nature, it will be shown that it can be used to find optimal solutions for several instances of the XHSTT archive ALL_INSTANCES. For those instances where an optimal solution cannot be found, we are able to show a non-trivial lower bound on optimum in the majority of cases. These are significant results for high school timetabling in general.

2 Mixed Integer Programming Formulation

In this section a brief description of the different terms of the MIP model is given.

S. Kristiansen (E-mail: sikr@dtu.dk) · M. Sørensen
MaCom A/S, Vesterbrogade 48, 1., DK-1620 Copenhagen V, Denmark

T.R. Stidsen
Section of Operations Research, Department of Management Engineering, Technical University of Denmark
Building 426, Produktionstorvet, DK-2800 Kgs. Lyngby, Denmark

An instance of XHSTT consists of *times* (denoted \mathcal{T} in the following), *time groups* (denoted \mathcal{TG}), *resources* (denoted \mathcal{R}), *events* (denoted \mathcal{E}), *event groups* (denoted \mathcal{EG}) and *constraints* (denoted \mathcal{C}). An event $e \in \mathcal{E}$ has a duration $D_e \in \mathbb{N}$, and a number of *event resources* which we each denote $er \in e$. An event resource defines the requirement of the assignment of a resource to the event, and this resource can be specified to be preassigned. If the resource is not preassigned, a resource of proper *type* must be assigned. Furthermore an event resource er can undertake a specific *role_{er}*, which is used to link the event resource to certain constraints.

It is the job of any solver for XHSTT to decide how each event should be split into *sub-events*. A sub-event se is defined as a fragment of a specific event $e \in \mathcal{E}$, has a duration $D_{se} \leq D_e$, and inherits the requirement of resources defined by the event, such that each sub-event has the exact same resource requirements as the event. Let \mathcal{SE} denote the entire set of sub-events, and let $se \in e$ specify that sub-event se is part of event e . The total duration of all sub-events for event $e \in \mathcal{E}$ in a solution cannot exceed D_e . In our model formulation we create the 'full set' of sub-events with different lengths, i.e. all possible combinations of sub-events for a given event can be handled. E.g. if an event has duration 4, the set of sub-events for this event has the respective lengths 1, 1, 1, 1, 2, 2, 3 and 4. As a constraint it is then specified that the summed duration of the *active* sub-events in a solution must equals 4. A sub-event is active if it is assigned a starting time or a non-preassigned resource. An active sub-event is analogous to the concept of *solution events* defined in the XHSTT documentation.

The times \mathcal{T} are ordered in chronological order, and we let $\rho(t)$ denote the index number of time t in \mathcal{T} . A time group \mathcal{TG} defines a set of times, and we let $t \in tg$ denote that time t is part of time group tg .

Each constraint $c \in \mathcal{C}$ is of a specific type, and the set \mathcal{C} can contain several constraints of the same type. Each constraint applies to certain events, event groups or resources, and penalizes certain characteristics of the timetable for these entities.

More details on the MIP model will be given in a full paper on the research.

3 Results

We have two primary intentions with our computational results:

- How does the MIP compete with the heuristics of the ITC2011 round 2? Thereby the potential of this MIP approach can be evaluated on fair terms with well-performing heuristics.
- Are we able to improve the best-known solutions for some instances, or even solve them to optimality?

All tests were run on a machine with an Intel i7 CPU clocked at 2.80 GHz and 12GB of RAM, running Windows 8 64 bit. In all cases the commercial state-of-art MIP solver Gurobi 5.5.0 was used. Two distinct sets of XHSTT instances have been used, both obtained from the XHSTT website (Post, 2013). All obtained solutions have been verified as being valid using the evaluator *HSEval* (Kingston, 2013).

An XHSTT objective consists of both a hard cost, and a soft cost, denoted (hard cost, soft cost). In case a solution has a hard cost of value 0, the objective is simply written as the soft cost, as is usually done in context of the XHSTT format. By this definition, hard constraints always take priority over soft constraint. Hence, we use a *lexicographic multi-objective optimization* technique, where we first solve problem containing only the hard constraints. In case of optimality the soft constraints are added, and a constraint is added to ensure the hard cost optimality.

Table 1 shows the obtained results of the comparison with the competitors of ITC2011. The value of "Avg. Ranking" was calculated as follows. Each solution method was ranked 1 to 5 on each instance, 1 being the best, and the average of these ranks was taken. According to this measure, the exact method of this paper is competitive with the methods used at ITC2011. On two instances the exact method gave the best results.

In attempt to produce new (optimal) solutions, the XHSTT archive ALL_INSTANCES¹ was used, which contains 38 non-artificial instances. According to the website, this archive "contains all latest versions of the contributed instances". The instances with a solution cost of 0 are skipped. Gurobu

¹ The archive and the results compared with are dated to 5th September 2013

was allowed to use all CPU cores (8 in our case) and a time-limit of 24 hours. Table 2 shows the obtained results. We were able to solve 4 instances was solved to optimality, proving optimality of 3 previously known solutions and finding 1 new optimal solution. Furthermore, 11 new non-trivial lower bounds and 7 new best solutions have been established for the instances which were not solved to optimality.

Table 1: Performance of the MIP using same running time as specified in ITC2011. For each instance is listed the average solution found from each of the competitors of ITC2011, and the solution obtained by the MIP formulations. The best solutions are marked in **bold**.

Instance	GOAL	HySST	Lectio	HFT	Exact method
BrazilInstance2	(1, 62)	(1, 77)	38	(6, 190)	46
BrazilInstance3	124	118	152	(30, 283)	39
BrazilInstance4	(17, 98)	(4, 231)	(2, 199)	(67, 237)	(5, 286)
BrazilInstance6	(4, 227)	(3, 269)	230	(23, 390)	682
ElementarySchool	4	(1, 4)	3	(30, 73)	3
SecondarySchool2	1	23	34	(31, 1628)	(1604, 3878)
Aigio	13	(2, 470)	1062	(50, 3165)	(1074, 3573)
Italy_Instance4	454	6926	651	(263, 6379)	17842
KosovaInstance1	(59, 9864)	(1103, 14890)	(275, 7141)	(989, 39670)	(3626, 2620)
Kottenpark2003	90928	(1, 56462)	(50, 69773)	(209, 84115)	(8491, 6920)
Kottenpark2005A	(31, 32108)	(32, 30445)	(350, 91566)	(403, 46373)	(2567, 53)
Kottenpark2008	(13, 33111)	(141, 89350)	(209, 98663)	-	(14727, 5492)
Kottenpark2009	(28, 12032)	(38, 93269)	(128, 93634)	(345, 99999)	(17512, 140)
Woodlands2009	(2, 14)	(2, 70)	(1, 107)	(62, 338)	(1801, 705)
Spanish school	894	1668	2720	(65, 13653)	(1454, 11020)
WesternGreece3	6	11	(30, 2)	(15, 190)	25
WesternGreece4	7	21	(36, 95)	(237, 281)	81
WesternGreece5	0	4	(4, 19)	(11, 158)	15
Avg. Ranking	1.72	2.67	2.50	4.44	3.61

4 Conclusion

Establishing optimal solutions and lower bounds is indeed a step forward for research within high school timetabling, and for the XHSTT format in particular. This gives researchers a possibility to compare their obtained solutions with an (optimal) lower bound, which is valuable for evaluating the quality of solutions.

References

- Bixby RE (2012) Optimization Stories, 21st International Symposium on Mathematical Programming Berlin, vol Extra, Journal der Deutschen Mathematiker-Vereinigung, chap A Brief History of Linear and Mixed-Integer Programming Computation, pp 107–121
- Kingston JH (2013) The hseval high school timetable evaluator. <http://sydney.edu.au/engineering/it/jeff/hseval.cgi> [Retrieved 12/8-2013]
- Kristiansen S, Stidsen TR (2013) A comprehensive study of educational timetabling - a survey. Tech. Rep. 8, 2013, DTU Management Engineering, Technical University of Denmark
- Post G (2013) Benchmarking project for (high) school timetabling. <http://www.utwente.nl/ctit/hstt/> [Retrieved 12/8-2013]
- Post G, Ahmadi S, Daskalaki S, Kingston J, Kyngas J, Nurmi C, Ranson D (2012) An xml format for benchmarks in high school timetabling. Annals of Operations Research 194:385–397
- Schaerf A (1999) A survey of automated timetabling. Artificial Intelligence Review 13:87–127

Table 2: Performance of the MIP on *ALL_INSTANCES*. For each instance is listed the best previously known solution "Best", and for the solution found by our approach is listed the time used to solve Step 1 "Time₁", the time used to solve Step 2 "Time₂". "Time" indicates the total solving time. All times have seconds as unit. Furthermore the objective "Obj" and the lower bound "LB" is listed. The percentage gap between the objective and the lower bound is divided into the gap for the hard constraints "Gap₁" and the gap for the soft constraints, "Gap₂". Objectives in **bold** denote new best solution while optimal solutions are marked with *.

		MIP solution method							
	Instance	Best	Time ₁	Time ₂	Time	Obj	LB	Gap ₁	Gap ₂
AU	BGHS98	(3, 494)	>86400	-	>86400	(3, 494)	(-, -)	-	-
AU	SAHS96	(8, 52)	>86400	-	>86400	(8, 52)	(-, -)	-	-
AU	TES99	(1, 140)	>86400	-	>86400	(1, 140)	(0, -)	100.0	-
BR	Instance1	42	0	>86400	>86400	40	28	0.0	30.0
BR	Instance2	5	1	>86399	>86400	5	1	0.0	80.0
BR	Instance3	47	1	>86399	>86400	26	19	0.0	26.9
BR	Instance4	78	1	>86399	>86400	61	42	0.0	31.2
BR	Instance5	43	1	>86399	>86400	30	10	0.0	66.7
BR	Instance6	60	1	>86399	>86400	60	14	0.0	76.7
BR	Instance7	122	1	>86399	>86400	122	22	0.0	82.0
DK	Falkoner2012	(2, 23705)	>86400	-	>86400	(2, 23705)	(0, -)	100.0	-
DK	Hasseris2012	(293, 32111)	>86400	-	>86400	(293, 32111)	(-, -)	-	-
DK	Vejen2009	(20, 18966)	>86400	-	>86400	(20, 18966)	(2, -)	90.0	-
UK	StPoul	136	52	>86348	>86400	136	0	0.0	100.0
FI	ElementarySchool	3	2	785	787	*3	3	0.0	0.0
FI	HighSchool	1	1	>86399	>86400	1	0	0.0	100.0
FI	SecondarySchool	88	1	>86399	>86400	88	77	0.0	12.5
GR	UniInstance3	5	0	3	3	*5	5	0.0	0.0
GR	UniInstance4	8	1	>86399	>86400	8	0	0.0	100.0
IT	Instance1	12	1	4561	4562	*12	12	0.0	0.0
IT	Instance4	78	12	>86389	>86400	62	27	0.0	56.5
XK	Instance1	3	31	>86369	>86400	3	0	0.0	100.0
NL	GEPRO	(1, 566)	>86400	-	>86400	(1, 566)	(0, -)	100.0	-
NL	Kottenpark2003	1410	57	>86343	>86400	1410	(0, -)	0.0	-
NL	Kottenpark2005	1078	88	>86312	>86400	1078	9	0.0	99.2
NL	Kottenpark2009	9250	92	>86308	>86400	9035	160	0.0	98.2
ZA	Woodlands2009	2	22	77878	77900	*0	0	0.0	0.0
ES	School	(3, 5966)	6525	>79875	>86400	357	322	0.0	9.8